

Orkestracija

- ❖ Koncept orkestracije
- ❖ Kubernetes

Automatizacija sistema velike skale (1/3)

- ◆ Pristup automatizacijom ručnih procesa:
 - Vodi ka deobi funkcionalnosti, tj. zasebnih alata
 - Alati za raspoređivanje, alati za konfigurisanje, itd.
- ◆ Potreban sistem za rukovanje svim aspektima
- ◆ Ova potreba je pojačana pojavom kontejnera:
 - Brzo stvaranje
 - Kratak životni vek
 - Replikacija
- ◆ Brzo stvaranje:
 - Vreme ručnog stvaranja je nedopustivo dugo

Automatizacija sistema velike skale (2/3)

◆ Kratak životni vek:

- Kontejner liči na app proces, brzo se završi i ugasi

◆ Replikacija:

- Čim se potražnja za servis poveća, pokreću se dodatni kontejneri za tu uslugu
- Kad se opterećenje smanji, suvišni kontejneri se gase
- Replikacija implicira i konfigurisanje mreže i skladišta

◆ Def. Orkestrator je:

- Automatizovan sistem za koordinaciju podsistema za konfiguraciju, raspoređivanje, vođenje i nadzor usluga

Automatizacija sistema velike skale (3/3)

- ◆ Dodatno Orkestrator obezbeđuje:
 - Dinamičko skaliranje usluga (kroz replikaciju)
 - Koordinacija više usluga
 - Otpornost na otkaze i automatski oporavak
- ◆ Koordinacija više usluga:
 - Orkestrator nadzire perfo. i uravnotežuje opterećenje replikacijom na manje opterećene servere
- ◆ Otpornost na otkaze i automatski oporavak:
 - Restart kontejnera u otkazu, ili
 - Uključivanje rezervnih (backup) kontejnera

Kubernetes (K8s, Kates)

- ◆ Kubernetes je orkestrator za rukovanje:
 - raspoređivanjem i vođenjem usluga na kontejnerima
 - Sledi tabela sa 7 glavnih funkcija K8s-a:

Rbr.	Osobina
1	Imenovanje i otkrivanje usluga
2	Uravnoteženje opterećenja
3	Orkestracija skladišta podataka
4	Optimizacija postavljanja kontejnera
5	Automatski oporavak
6	Rukovanje konfiguracijama i zaštitom
7	Automatizovano uključanje nove/stare verzije SW (rollout/rollback)

K8s funkcije (1/3)

◆ Imenovanje i otkrivanje usluga:

- Usluzi se može pristupiti preko domena ili IP adrese

◆ Uravnoteženje opterećenja:

- Po potrebi, K8s stvara više kopija kontejnera
- Koristi balanser opterećenja za raspoređivanje zahteva na te kopije

◆ Orkestracija skladišta podataka

- Automatsko motiranje skladišta pri pokretanju kontejnera
- Skladišta: lokalna, na javnim oblacima, itd.

K8s funkcije (2/3)

◆ Optimizacija postavljanja kontejnera:

- Usluga radi na klasteru servera (čvorova)
- Operater specificira procesor i RAM za kontejer
- K8s postavlja kontejnere na čvorove tako da optimizuje upotrebu servera

◆ Automatski oporavak:

- K8s zamenjuje kontejner koji je otkazao
- K8s zatvara kontejner koji je prestao da odgovara na periodične poruke provere ispravnosti (health-check)

K8s funkcije (3/3)

- ◆ Rukovanje konfiguracijama i zaštitom:
 - Informacija o konfiguraciji je razdvojena od slike
 - Konfiguracije: mreže, skladišta, itd.
 - Čuvanje osetljive info: lozinke, autentifikacioni tokeni, ključevi za šifrovanje
- ◆ Automatsko uključivanje nove/stare verzije:
 - Korisnik pravi sliku nove verzije usluge
 - K8s zamenjuje tekuće kontejnere sa novom verzijom (zaustavi postojeći i pokrene kontejner za zamenu)
 - Novi kontejner nasleđuje sve resurse starog

K8s ograničenja (1/2)

◆ K8s ne obezbeđuje:

- Fokus na specifične tipove app
- Rukovanje izvornim kodom i pravljenje slika
- Midlver za prosleđivanje događaja
- Nadzor i zapisivanje događaja

◆ Fokus na specifične tipove app:

- K8s je nezavistan od tip app; ne obezbeđuje spec fije

◆ Rukovanje izvornim kodom i pravljenje slika:

- K8s se oslanja na tehnologiju kontejnera, npr. doker
- Podrazumevano: OS Linux sa dockerd i docker UI

K8s ograničenja (2/2)

◆ Midlver za prosleđivanje događaja:

- Paradigma programiranja na bazi događaja zahteva slanje asinhronih događaja između kontejnera
- K8s ne obezbeđuje takav midlver

◆ Nadzor i zapisivanje događaja:

- K8s nadzire i upravlja kontejnerima, ali
- Ne obezbeđuje mehanizme za skupljanje, zapis, i izveštavanje o merenjima
- Postoje samo neka opciona početna rešenja

Model kubernetes klastera (1/2)

◆ K8s ima mnogo komponenata:

- Koje se mogu konfigurisati na puno raznih načina
- Model objašnjava mesto i ulogu komponenata

◆ K8s klaster je skup kontejnera:

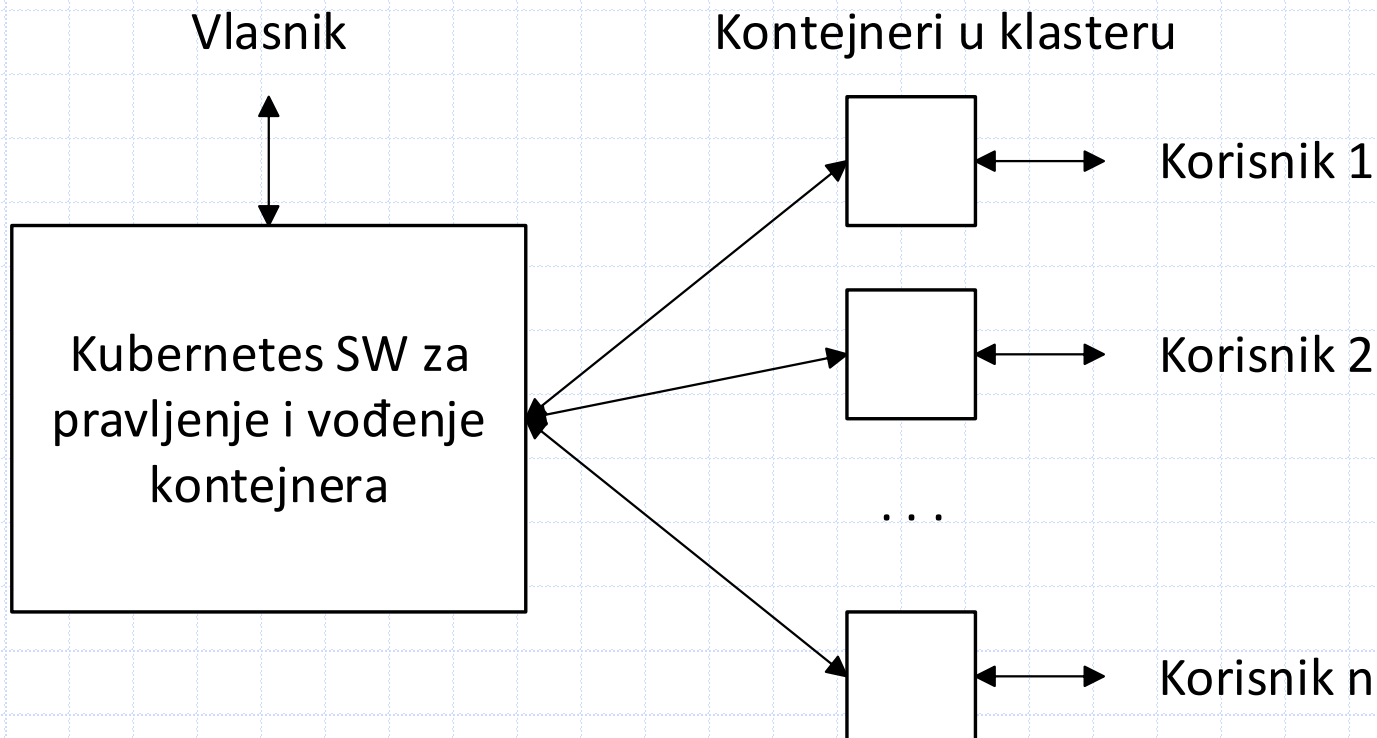
- Plus SW za pravljenje/vođenje/pristup kont.
- Br kont. se povećava/smanjuje prema opterećenju

◆ SW u klasteru se deli u 2 kategorije:

- SW (za vlasnika) za pravljenje i vođenje kontejnera
- SW (za korisnika) za pristup kontejneru
- Vlasnik/korisnik mogu biti i programi

Model kubernetes klastera (2/2)

- ◆ Obe kategorije SW nude više API-ja:
 - Npr. I kat. obezbeđuje CLI i RESTful, a II Web



Kubernetes pod

- ◆ U K8s sinonim za app je „pod“
- ◆ K8s raspoređuje skup aplikacija (tj. pod-ova):
 - Mnoge app samo na jednom kontejneru
 - App na bazi mikroservisa se sastoje od više delova, koji se izvršavaju na posebnim kontejnerima
- ◆ U K8s pod je najmanja jedinica rada
 - K8s raspoređuje sve kontejnere pod-a na isti čvor
- ◆ K8s dodeljuje 1-nu IP adresu celom pod-u:
 - Kontejneri pod-a komuniciraju preko localhost sprege
 - Zato neki TCP port sme koristiti samo jedan pod

Pravljenje pod-a, mustre, i vremena povezivanja

◆ Pravljenje pod-a:

- K8s koristi tzv. kasnije-povezivanje (late binding)
- Programer pravi tzv. mustru za pod

◆ Mustra imenuje pod i specificira:

- Kontejnere koje treba pokrenuti
- Spisak mrežnih port-ova
- Verziju kubernetes API-ja

◆ Mustra može biti u yaml ili json formatu

- Primer za pod koji radi na jednom kontejneru i koristi port 8080

Primer mustre

```
apiVersion: v1
kind: Pod
metadata:
  name: example-pod
  labels:
    purpose: web-server
spec:
  containers:
  - name:example-pod
    image: f34cd9527ae6
    ports
      - containerPort:8080
```

- Info iz mustre se čuva u instanci pod-a
- Tu info koristi K8s da kontroliše i vodi pod
- Ta info može sadržati i promenljive okruženja, tačke montiranja spoljnog skladišta, i inicijalne komande koje treba izvršiti

Inicijalni (init) kontejneri

- ◆ Pod može da ima 1 ili više init kontejnera:
 - Oni rade inicijalizaciju pre pokretanja glavnih kont.
- ◆ Init kont se najčešće koriste za provere, npr:
 - Da li je eksterno skladište montirano ili ne?
 - Da li su raspoloživi repozitorijumi i spoljne računske usluge koje pod koristi?
- ◆ Zašto se te provere ne rade u glavnim kont.?
 - Neki kontejneri bi mogli da uredno startuju, a neki ne zato što im nešto nedostaje
 - Takav nepotun pod ne bi mogao da radi ispravno

Čvorovi i Kontrolna ploča

- ◆ Def. Čvor = računski čvor, tj. računar/VM:
 - K8s raspoređuje kontejnere na čvorove
- ◆ Def. Kontrolna ploča (ili Master):
 - SW (za vlasnika) za pravljenje i vođenje kontejnera
 - Čvor na kom se izvršava se zove Vodeći (Master) čvor
- ◆ Def. Čvor radnik (ili K8s čvor):
 - Je čvor na kom se izvršavaju kontejneri pod-a
- ◆ Moguće pokrenuti više kopija kontrolne ploče
 - Koje rade na više vodećih čvorova
 - Međutim, obično postoji 1 kopija na 1 vodećem čvoru

Komponente Kontrolne ploče (1/2)

- ◆ Kontrolna ploča sadrži sledećih 5 komponentata
- ◆ API server (kube-api-server):
 - Obezbeđuje interfejs prema svim komponentama
- ◆ Raspoređivač (kube-scheduler):
 - Raspoređuje pod-ove na čvorove klastera
 - Stalno radi pa može da vrši dinamičku dodelu
- ◆ Skladište stanja klastera sadrži info o klasteru:
 - Koristi se etcd – dist. skladište više kopija parova ključ-vrednost, koje koristi alg. konsenzusa za pouzdano očitavanje vrednosti za zadati ključ

Komponente Kontrolne ploče (2/2)

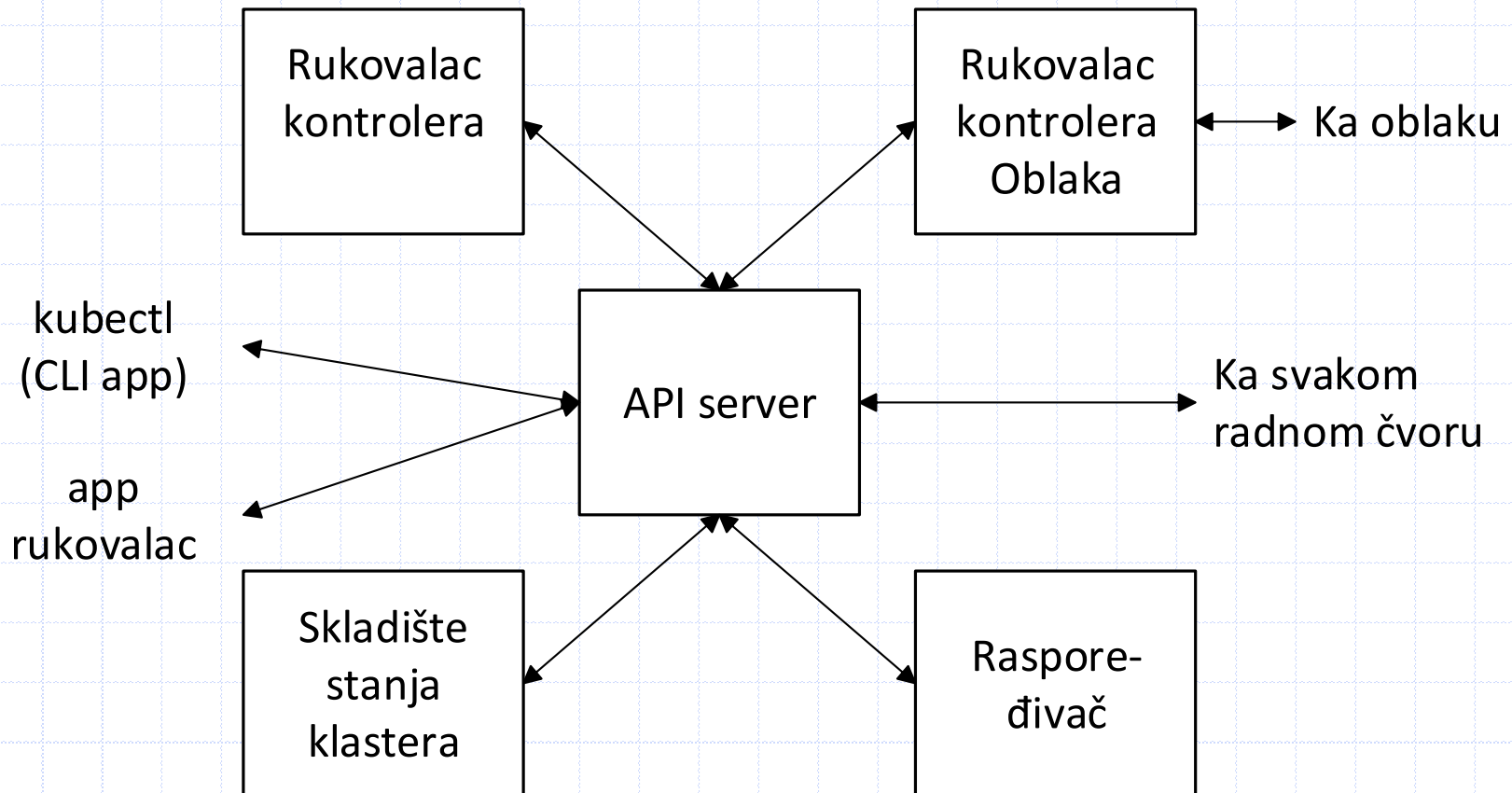
- ◆ Rukovalac kontrolera (kube-controller-manager)
 - Demon koji izvršava skup kontrolnih petlji (kontrolera)
 - Kontroleri za: replikaciju, krajnje tačke, prostor imena
 - Skupljanje sl.mem., završavanje pod-ova i čvorova
- ◆ Rukovalac kontrolera oblaka:
 - Obezbeđuje interfejs prema korišćenom oblaku
 - Npr. slanje zahteva i proba
 - Zahtevi za: izmene mrežnih putanja, montiranje skladišta, proveru stanja HW
- ◆ Kubectl CLI app obezbeđuje vezu API serverom

Komunikacija između komponenata kontrolne ploče (1/2)

- ◆ Pret. da postoji 1 vodeći čvor
- ◆ Komunikacija:
 - Sve komponente komuniciraju između sebe
 - Rukovalac kontrolera oblaka komunicira i sa oblakom
 - API server ima ulogu unutrašnjeg komutatora

Komunikacija između komponenata kontrolne ploče (2/2)

Komponente vodećeg (master) čvora



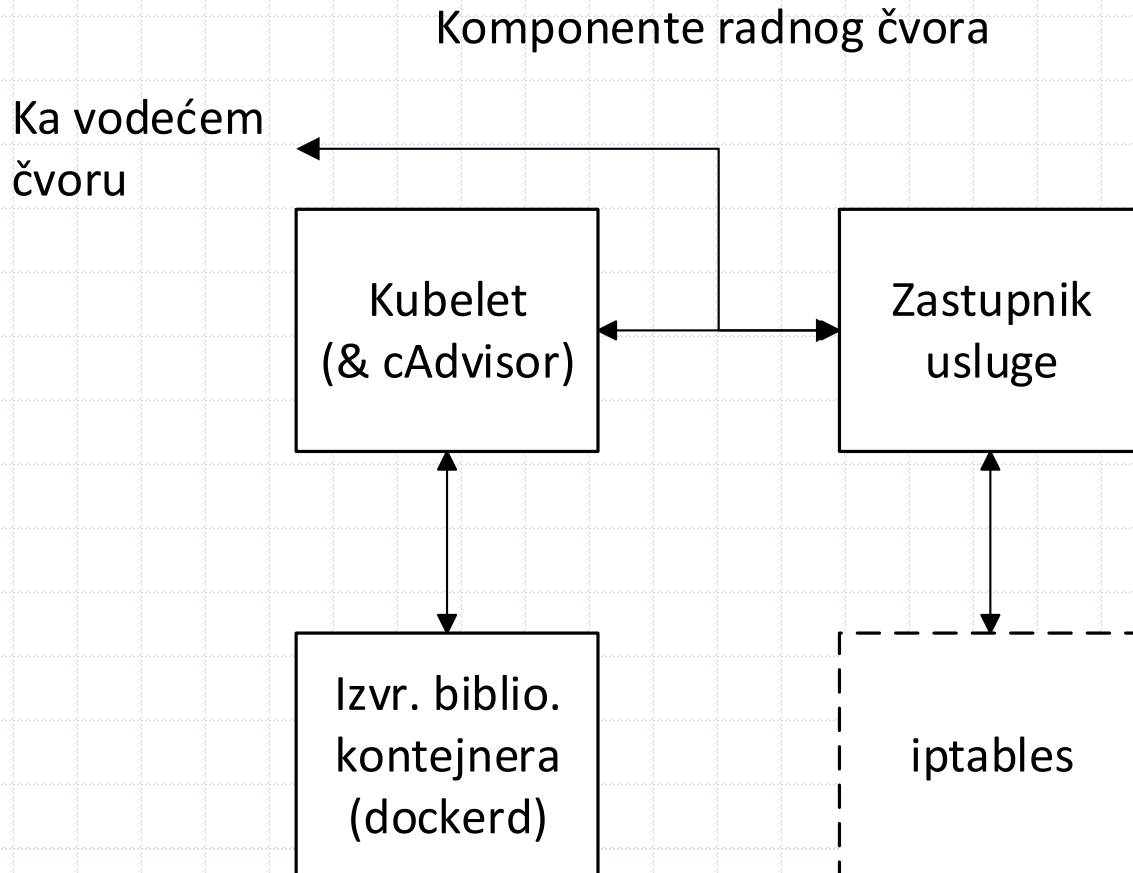
Komponente radnog čvora (1/3)

- ◆ Postoje sledeće 3 komp. za vođenje pod-ova
- ◆ Zastupnik usluge (service proxy):
 - Konfigurise mrežna usmeravanja na čvoru
 - Konkretno, konfigurise Linux iptables
- ◆ Kubelet
 - Spreže kontrolnu ploču i radni čvor
 - Postavlja izolovano okruženje za pod
 - Koristi SW kontejnera za izvršenje i nadzor pod-a
 - Koristi SW cAdvisor za skupljanje statistike o pod-ima
 - Šalje izveštaj kroz Summary-API ka Metrics serveru

Komponente radnog čvora (2/3)

- ◆ Izvršna biblio. kontejnera (container runtime):
 - Može se koristiti bilo koji sistem kontejnera
 - K8s podrazumevano koristi docker
 - Kad K8s treba da rasporedi kontejnere, on taj posao delegira izvršnoj biblioteci kontejnera

Komponente radnog čvora (3/3)



Dodatne funkcije K8s (1/4)

◆ Replike:

- Prilikom raspoređivanja klastera, vlasnik može da eksplicitno kontroliše br. replika, tj. skaliranje app

◆ Rasporedi (deployments):

- K8s prati pristup na bazi namere (intent)
- Korisnik zadaje broj replika, a K8s automatski održava zahtevani broj

◆ Skup app sa stanjem (StatefulSets):

- Skup pod-ova sa zagaranovanim redosledom
- Svaki pod ima jedinstven ID, koji je perzistentan

Dodatne funkcije K8s (2/4)

- ◆ Skup demona (DaemonSet):
 - Mogu se pokrenuti kopije pod-a na svim čvorovima
 - To su demoni koje koriste drugi kontejneri na čvoru
- ◆ Skupljač oslobođene memorije (garbage):
 - K8s objekti imaju vlasnike
 - Kad se vlasnik završi, skupljač sakupi oslobo. mem.
- ◆ TTL kontroler (TTL controller):
 - TTL: Max vreme života objekta nakon završetka
 - Kad TTL istikne, TTL kontroler javlja skupljaču o.m.

Dodatne funkcije K8s (3/4)

◆ Posao (Job):

- Pokreće zadat br. pod-ova i prati njihov napredak
- Ako neki ispadnu pre tog br., Posao ih restartuje
- Npr. garancija da će se 1 pod sigurno završiti

◆ Periodičan posao (CronJob):

- Može se rasporediti periodičan posao (ala Unix cron)

◆ Usluge (Services):

- Skup pod-ova sa zadatom politikom pristupa
- Dolazeći zahtevi se raspoređuju na pod-ove
- Ispadi/restarti pod-ova ne utiču na app koja ih koristi

Dodatne funkcije K8s (4/4)

◆ Usluge - nastavak:

- Osnova za arhitekture sa mikroservisima

◆ Za mnoge zadatke K8s nudi 2 pristupa:

- Izdavanje eksplicitnih detaljnih komandi, ili
- Korišćenje automatizovanog sis. koji rukuje detaljima